IDENTIFICATION OF α-HELICES FROM LOW RESOLUTION PROTEIN DENSITY MAPS

A. Dal Palù
Dip. Matematica
Università di Parma
alessandro.dalpalu@unipr.it

J. He, E. Pontelli*, Y. Lu
Dept. Computer Science
New Mexico State University
{*jinghe,epontell,ylu*}*@cs.nmsu.edu*

This paper presents a novel methodology to analyze low resolution (e.g., 6Å to 10Å) protein density map, that can be obtained through electron cryomicroscopy. At such resolutions, it is often not possible to recognize the backbone chain of the protein, but it is possible to identify individual structural elements (e.g., α-helices and β-sheets). The methodology proposed in this paper performs *gradients* analysis to recognize volumes in the density map and to classify them. In particular, the focus is on the reliable identification of α-helices. The methodology has been implemented in a tool, called *Helix Tracer*, and successfully tested with simulated structures, modeled from the Protein Data Bank at 10Å resolution. The results of the study have been compared with the only other known tool with similar capabilities (*Helixhunter*), denoting significant improvements in recognition and precision.

1. INTRODUCTION

3-dimensional (3D) protein structure information is essential in understanding the mechanisms of biological processes. A protein can be thought of as a chain of beads that adopts a certain conformation in the 3D space (*native conformation*). The building blocks of the chain are 20 kinds of amino acids. Knowledge of 3D structure of proteins is essential in understanding the mechanisms of protein function, and this information has become more and more important in rational drug design.

Both *experimental* techniques and *prediction* techniques have been devised to generate 3D protein structures. The most commonly used experimental techniques for protein structure determination are *X-ray crystallography* and *Nuclear Magnetic Resonance (NMR)*. Both techniques can determine structures at atomic resolution (usually better than 3Å). In particular, X-ray crystallography has produced more than 80% of the known protein structures currently present in the *Protein Data Bank (PDB)*. Although these two techniques are successful in targeting soluble proteins, they are seriously limited for non-soluble proteins, such as membrane bond proteins and large protein complexes. In particular, X-ray crystallography is limited to the availability of suitable crystals of the protein, and large protein complexes cannot easily produce crystals.

Electron cryomicroscopy is an experimental technique that has the potential to allow structure determination for large protein complexes [15, 12, 2]. Using the cutting edge techniques in this field, 3D structure of large complexes, such as the Herpes virus, have been successfully generated at 8.5Å resolution [15]. Although it is not possible to determine the backbone chain of the protein at the resolution range of 6Å to 10Å—current methods to solve protein structure require a density map of much higher resolution, such as 3Å or 4Å [14, 6]—this resolution allows the visualization of various secondary structure elements, such as α-helices and β-sheets [15].

In this paper, we present a new methodology to aid in the identification of α-helices in a low resolution density map. The methodology relies on a novel representation of α-helices, where helices are modeled as general cylinder-like shapes, defined by a central axial line (i.e., a *spline*). The spline is

*Corresponding author.

a continuous line (possibly *not straight*), described by a set of control points. This feature allows the model to better fit real helices, and thus provides smaller errors, since helices in nature are often not straight. The actual identification of the helices makes use of a new type of analysis of the density maps, based on the notion of *gradient segmentation*. The strength of this segmentation method is its threshold independence—which allows the segmentation of volumes present in the density map without the drawback of using generic thresholds, that can be inadequate for specific regions of the density map. The segmentation we propose is *general*, and can be potentially used for extraction of other features, e.g., β-sheets and coils. The proposed methodology has been implemented in a software tool, called *Helix Tracer*. Preliminary experiments show very promising results—*Helix Tracer* is on average capable of identifying over 75% of the helices, with very low RMSD errors, and with greater accuracy than related systems (the *Helixhunter* system [7]).

To the best of our knowledge, only one other approach has been proposed to identify helices in low resolution density maps: *Helixhunter* [7] relies on a cylindrical representation of helices, where each helix is described as a *straight* cylinder with a 5Å diameter. *Helixhunter* identifies helices by searching cylindrically shaped areas in the density map using the second moment tensor.

Since α-helices and β-sheets are the major components of a structure, the knowledge of this information helps in discovering the fold of a protein. Moreover, these secondary structure components help in producing important geometrical constraints about the tertiary structure. Such constraints can be employed to effectively guide a protein prediction method, significantly improving precision and efficiency of the prediction [4], as well as to reduce the search space in the context of molecular dynamics applications. For example, in [4], we present an effective method, based on constraint satisfaction, to combine information about α-helices, obtained from *Helix Tracer*, with results from helix prediction (obtained from PHD [13]), with the aim of determining the most likely mappings of the α-helices on the primary sequence.

2. METHOD

The input to our analysis algorithm is a *density map*, encoded as a 3-dimensional array. Each element corresponds to a cubic volume, called *voxel*, and each voxel is associated to the mean electron density of the protein in that volume. For the sake of simplicity, the density is normalized w.r.t. a maximal density in the map.

Our analysis method relies on the observation that, at the resolution range of 6Å–10Å [15, 7], it is possible to observe that the density distribution of a helix resembles a cylinder. In particular, the cylindrical area presents the local maximum density value roughly on the central axis of the corresponding α-helix. The density gradually decreases as the distance from the central axis increases. However, most helices have a certain degree of curvature, particularly for long helices, thus making a perfect cylinder not an accurate template.

2.1. Overall approach

The algorithm for helix extraction is based on processing the discrete density map. The outcome of this analysis is a description of the helices identified. In various previous proposals, such as in *Helixhunter* [7], each α-helix is described as a cylinder with a 2.5Å radius. The cylinder is characterized by three parameters (see Fig. 1 on the left): *(a)* the *starting point*, $\vec{s} = (s_x, s_y, s_z)$, located on one extremity of the central axis of the cylinder, *(b)* the axis orientation vector $\vec{d} = (d_x, d_y, d_z)$, and *(c)* the length of the axis ℓ. Following our concern, that actual helices in nature are not straight but they tend to bend and curve to a certain degree, we introduce a more general representation. In this work, we describe the central axial line of the α-helix in terms of a *quadratic spline* [1], while the helix itself is defined as the set of points whose minimal distance from the spline is 2.5Å. A spline is a continuous curve, controlled by a finite number of control points. The central axial spline is generated using a standard spline function, based on the identified control points $\vec{a}_1 \ldots \vec{a}_n$, where $\vec{a}_i = (a_{ix}, a_{iy}, a_{iz})$—see Figure 1 on the right.

The essential idea used in the helix detection process is to segment the density map into *volumes* satisfying certain properties.

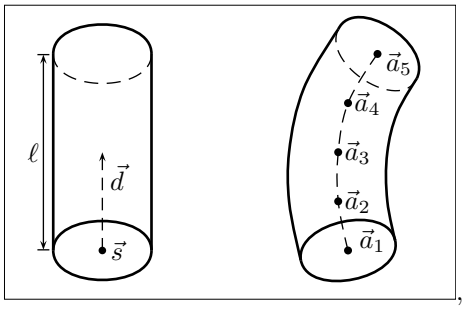


Fig. 1. Helix models

The intuition of the segmentation process is that each local maximal density voxel can be related to the presence of a packed set of atoms. This situation arises when amino acids are arranged into specific patterns that provide a high local density contribution. For example, helices are arranged so that the side chains of amino acids involved show an average increase of local density w.r.t. normal coil, due to the helical packing of the backbone. At low resolution, this is characterized by a clear increase of local density that reflects the helical three dimensional shape of the helix. Hence, the problem boils down to recognize such clusters made of locally higher density.

Every maximal density voxel v is a representative of a volume that is defined as the set of voxels that can be reached from v without increasing the density along the path followed. Each volume is a maximal set of voxels and it contains, in general, small parts of individual helices. The key idea is that this segmentation offers a robust identification of subsets of helices' volumes. Thus, the problem boils down to the one of correctly merging some of these volumes in order to reconstruct the identified helices.

The method involves *gradient* analysis, and it is substantially different from simple density values thresholding (as used in previous proposals [7, 8]). The gradient is a vectorial information, expressed in terms of a 3D direction and intensity. Intuitively, the gradient shows the direction that points to the locally maximal increase of density. The gradient information is computed for each voxel, considering the density map as the discretization of a continuous function from R^3 to density values. In this perspective, the gradient corresponds to the first derivative of such density function. For processing purposes, the gradient associated to each voxel is approximated using a discrete and local operator over the original density map.

Using the gradient direction as a *pointer*, we can follow these directions from voxel to voxel, until a *maximal* density value voxel is found. The paths generated touch every voxel, and can be partitioned according to the ending points reached. Paths that share the same ending point form a tree structure, that is associated to the same volume. This process generates in output the segmentation we require for helix detection.

The motivation for requiring such segmentation is that low resolution density maps witness the presence of a helix as a dense cylinder-like shape, where the maximal density increases gradually towards its axis. When close to the axis (e.g., ≤ 5Å), the gradient points towards such axis. This means that the high density voxels of the trees identified on the gradient paths can be employed to characterize the location of the helix axis. Observe that we use gradient trees to segment volumes—by collecting in a single volume all the nodes whose gradient paths lead to the same maximal density voxel. Thus, each of these volumes will contribute to only a part of a helix, and further analysis is required to study the properties of the volumes (and the relationships between their maximal density voxels) and determine whether different volumes actually belong to the same helix.

The complete process is articulated in the following phases, described in the next subsections: *(i)* gradients calculation, *(ii)* graph construction and processing; *(iii)* detection of helices.

2.2. Gradients determination

The density map is processed, in order to build the map of gradients. The gradient is approximated using Sobel-like convolution masks ($3 \times 3 \times 3$) over the original density map [5]. The gradient is represented by a vector whose direction and intensity can be calculated using the Sobel-like mask in Figure 2. For each voxel, a 3D convolution process is performed using the three masks: each mask is overlapped on the density map, and the summation of a point-by-point product is performed in order to collect the intensity of the gradient component for each dimension. The

addition of the three resulting vectors generates the gradient associated to the voxel. For example, the component of the gradient along the X axis can be calculated by using the three matrices in the first row in Figure 2.

In Fig. 3(a), we show a slice of a density map; Fig. 3(b) indicates the corresponding z-projection of the gradient for each point. Fig. 3(c) is the overlay of Fig. 3(a) and Fig. 3(b). Observe how the gradient lines are "pointing" towards the denser regions of the density map (shown in darker color).

2.3. Construction of the graph

The next step of the algorithm involves the construction of a graph describing the structure of the density map. In particular, the directed graph $G = (N, E)$ is used to summarize the gradient properties, where N is the set of nodes of the graph and $E \subseteq N \times N$ is the set of edges. Nodes will represent voxels of interest (as described later) while edges connect voxels that are "adjacent" in the density map.

Let us consider two voxels $V_1 = \langle x_1, y_1, z_1 \rangle$ and $V_2 = \langle x_2, y_2, z_2 \rangle$. The voxels are considered to be *neighbors* if and only if the following relationship is satisfied: $\max\{|x_1 - x_2|, |y_1 - y_2|, |z_1 - z_2|\} \leq 1$. In other words, two voxels are neighbors if they differ by at most by one unit in each coordinate—which leads to 26 possible neighbors per voxel. In the graph we propose to construct, edges will be introduced only if the two nodes involved are neighbors.

The process starts with a coarse thresholding (*cropping*) of the density map. The purpose of this step is to discard grossly irrelevant voxels, to improve efficiency of the successive analysis steps, without incurring in any relevant loss of information. In particular, we retain only the voxels with a density value greater than 0.5 (50% of the maximum value of the map); this choice arises from the practical observation that the voxels of interest have an average density larger than 0.65.

After this coarse thresholding, N consists of the nodes formed by the remaining voxels. For each node n_1, we add a direct edge (n_1, n_2) that starts from n_1 and points to the neighbor $n_2 \in N$ of n_1 (the arrowed lines in Figure 4) if the following two conditions are satisfied:

- The directed edge is the best approximation of the gradient direction (the non-arrowed lines in Figure 4);
- The density at n_2 is higher than that at n_1.

As last step in the construction, the direction of each edge in the graph is inverted. The resulting graph is a directed acyclic graph, and each node has at most one incoming edge (Figure 4(c)). The graph is actually a forest of tress, since it is not necessarily connected. The key property is that every path in each tree represents a decreasing density sequence of neighboring voxels.

The trees recognized in this graph construction provide a segmentation of the density map in distinct *volumes*. Each volume contains the voxels that belong to the same tree. The *root* of a tree is the representative of the associated volume, and it is the voxel with the highest density in the volume—e.g., the double-circled node in Figure 4(c).

After the trees have been generated, small and spatially close trees are merged to simplify the forest of trees. Note that in this version we did not apply any image preprocessing, i.e. smoothing and/or filtering. This implies that some noise could be present and split the gradient segmentation into small volumes. However, due to the low resolution scale (6–12Å), we can recover this problem my merging manually volumes that are close to each other. This is important, in order to reduce volumes fragmentation and to facilitate detection of volume borders. When the distance between the roots of two trees are less than 3.0Å, the two trees are merged, because the typical distance between consecutive amino acids is 3.8Å. In the cases we select, the roots are close enough to ensure that the merged volumes describe two areas that are consecutive according to the direction defined by the backbone. In the future we plan to introduce a more robust image preprocessing (e.g. a smoothing phase) to cope with this problem.

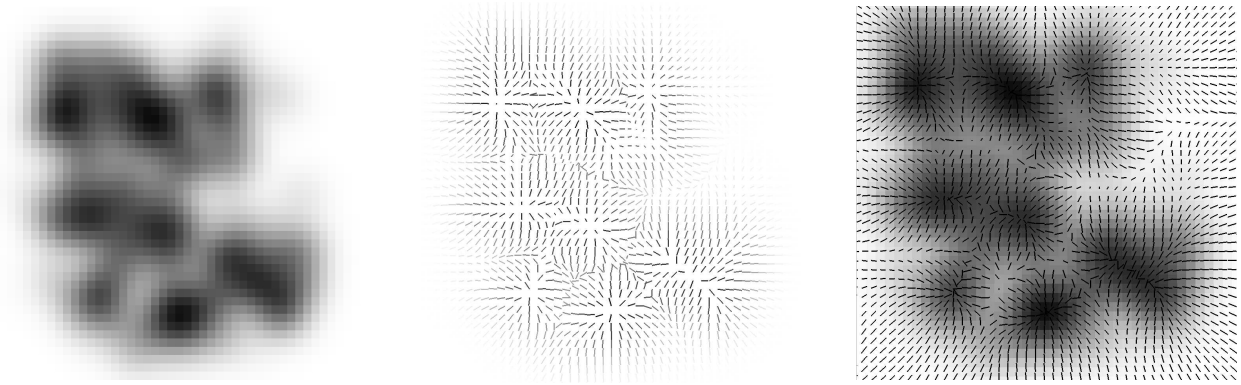
The last operation in the graph processing phase is to mark the border of each volume induced by the tree. A voxel is on the border if at least one of its neighbors is not in the same volume. The border voxels are used later for merging volumes that are determined to belong to the same helix. Voxels belonging to a volume border are properly marked using a flag.

$$Sobel_x[0] = \begin{bmatrix} 1 & 0 & -1 \\ 2 & 0 & -2 \\ 1 & 0 & -1 \end{bmatrix} \quad Sobel_x[1] = \begin{bmatrix} 2 & 0 & -2 \\ 4 & 0 & -4 \\ 2 & 0 & -2 \end{bmatrix} \quad Sobel_x[2] = \begin{bmatrix} 1 & 0 & -1 \\ 2 & 0 & -2 \\ 1 & 0 & -1 \end{bmatrix}$$

$$Sobel_y[0] = \begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix} \quad Sobel_y[1] = \begin{bmatrix} 2 & 4 & 2 \\ 0 & 0 & 0 \\ -2 & -4 & -2 \end{bmatrix} \quad Sobel_y[2] = \begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix}$$

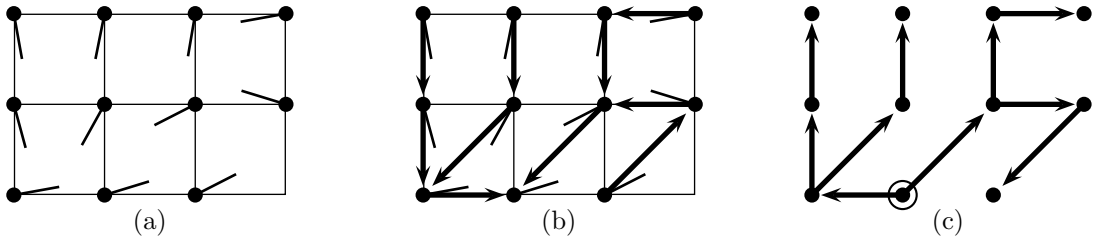
$$Sobel_z[0] = \begin{bmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{bmatrix} \quad Sobel_z[1] = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \quad Sobel_z[2] = \begin{bmatrix} -1 & -2 & -1 \\ -2 & -4 & -2 \\ -1 & -2 & -1 \end{bmatrix}$$

Fig. 2. Sobel's Convolution Masks



(a) (b) (c)

Fig. 3. Density map and gradients



(a) (b) (c)

Fig. 4. Obtaining the tree from gradients

2.4. Detection of helices

The final phase in our computation is to define the location of the α-helices. This phase involves two steps. The first step is to detect and merge the volumes that belong to the same helix. The second step is to construct a description of each identified helix, by defining the location of the control points that constitute the central axial line.

It has been observed that a helix often contains one or more neighboring volumes. Therefore, the volumes are analyzed to see if they belong to the same helix. The border voxels of each volume are scanned for the satisfaction of two rules. One is to see if there is a neighboring border voxel that belongs to a different volume. The other is to see if the border voxel has high density (greater than a threshold helixTHR). A typical threshold is 85% of the maximum density value of the map. Volumes that satisfy the

above two rules are merged to generate the volume of a helix. Note that this thresholding is applied *after* the segmentation phase, and thus is used simply as a selection tool for the relevant volumes.

The rationale of this process is the following. If two volumes belong to the same helix, this implies that the contact points between the two volumes lie on a plane which is orthogonal to the helix axis. In practice, due to the discretization of the density in voxels, the contact surface may not be a regular plane, and it may contain some irregularities. Nevertheless, this does not constitute a problem. This follows from the gradients property: each gradient associated to a voxel that belongs to a helix points towards the axis of the helix. Therefore, if the contact surface is not orthogonal to the axis, the gradients on the border will point to the volume, and thus one volume would be a subset of the other. This also implies that the high density region on the border between two volumes is very close to the axis of the α-helix. We have experimentally observed that only helices show a high density (above the *helixTHR* threshold) close to their axis, thus we can safely merge two volumes that presents this characteristic.

The identification of the control points relies on the fact that the central axial line of the helix is often located at the high density voxels of the volume. To define the search space of the control points, a subset of the helix volume, called *trace volume* is generated using a threshold (*helixselectTHR*). This threshold, by default, is set to 2% less than *helixTHR*. This choice is suggested by the practical observation that *helixTHR*, the threshold used to detect volumes belonging to same helix, is too strict when used for the construction of the axis. Moreover, note that this thresholding is performed on volumes, which ensures that we will not encounter cases where the analysis takes us to volumes that do not belong to the same helix. This guarantees that separate helices are not erroneously merged or a helix incorrectly broken in separate structures.

The central axial line is generated using a greedy algorithm. The idea is to start from the highest density voxel close to the barycentre of the trace volume (in the neighborhood of 3Å). We estimate the initial search direction by means of a least square fitting of the trace volume. From the starting voxel, we launch two searches along the initial search direction, that returns two half axis: one for each side analyzed. The traversal moves to a neighbor that contains the locally highest density available. During this exploration, we move along the axis towards the ends of the helix, while building a path that contains the maximal values detected; recall that the density map for a helix decreases quickly when moving orthogonally away from the axis. The union of the two paths gives the set of control points associated to the axis.

Finally, the control points are smoothed with a single pass of Gaussian smoothing ($\sigma = 8$), in order to reduce the scattered jumps between neighbors. The smoothed and real-coordinate points are used as the actual control points for the second-order spline that is consequently generated. At the end of the process, a validation step is launched, in order to discard the helices that show an extreme and unnatural curvature.

3. EXPERIMENTAL RESULTS

3.1. Helix Tracer results

In order to test Helix Tracer, we generated density maps for 29 proteins with known structures in the PDB. The density maps have been generated at 10Å resolution, using the program `pdb2mrc` from the EMAN suite [11]. The proteins have been randomly chosen, and they offer a good representation of proteins with varying structural configurations. In particular, we include representatives from the major *SCOP* families [10]—$\alpha + \beta$ (e.g., 1A06), all α (e.g., 1CC5), α/β (e.g., 1B0M), and proteins with separate α and β domains (e.g., 1BVP). For example, the density map of protein 1BM1, at 10Å resolution, is shown in Figure 5. The helices identified by *Helix Tracer* are shown as sticks that are overlayed on the density map and on the backbone of the protein. Notice that the helices identified are not straight. All experiments have been conducted using Linux (2.6.11) workstations (a Pentium 4, 3.1GHz, and a AMD 64-bit 2.39GHz).

Table 2 reports the number of helices recognized by *Helix Tracer*. In particular, Table 2 provides the following information: the PDB Id of each protein (**ID**), the execution time, in seconds, (**Time**), the number of helices present in the PDB annotation

(**PDB Helices**), the number of helices in the PDB annotation that are longer than 8.1Å[a], the number of helices recognized by *Helix Tracer* (**Identified**), the number of identified helices that are correct (**Correct**), the number of false positives (**False**), and the number of helices of adequate length present in the PDB annotation and missed by *Helix Tracer* (**Missed**). The last two columns report the number of C_α present in the helices of length greater than 8.1Å in the PDB annotation, and the corresponding number of C_α correctly identified by *Helix Tracer*.

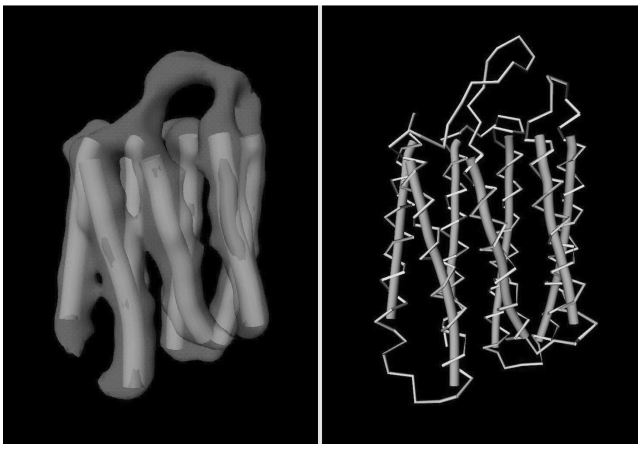


Fig. 5. Helices identified for 1BM1 (PDB Id)

A helix is correctly identified by *Helix Tracer* if it can be paired with a PDB helix in the protein. In particular, there should be at least one C_α on the PDB helix that is within a 4Å distance from the central axial line of the identified helix. *Helix Tracer*, on average, recognizes 80% of the helices longer than 8.1Å. In particular, often the ratio of helices correctly identified is larger than 88%.

Although the pairing process is simple, the accuracy of the identified helices can also be seen from the number of C_α that are recognized by *Helix Tracer*. A C_α is a correctly recognized C_α atom if it can be projected internally on the helix axis identified. C_α atoms that cannot be projected on the axis—i.e., they could be projected on the prolongation of the axis outside the helix—are not accounted as correctly identified C_α atoms. *Helix Tracer* recognized 75% of the total C_α atoms that are on the PDB helices longer than 5 amino acids (shown in Table 2 and in Figure 6 on the left, as comparison). Observe also that, despite the lack of optimization in the current implementation, the execution times are very reasonable (Table 2, column **Time**).

When a helix is represented as a straight cylinder, it is straightforward to calculate the projection of a C_α atom on the central helix axis. However, since we use splines as axis representation (see Figure 1), a method to project a C_α on the axial spline needs to be developed. We subdivide the continuous spline into a set of 40 contiguous segments, a sufficient number to approximate the spline. The lengths of these segments are not necessarily identical, and they depend on the spatial distribution of the control points. We approximate the problem of computing the projection on a continuous spline with the problem of finding the smallest projection vector out of the set of segments.

In order to further evaluate the accuracy, the *RMSD* (*Root Mean Square Deviation*) is calculated for the correctly identified C_α atoms. The theoretical distance between a C_α atom and the central axial line of the helix is 2.5Å. The *RMSD* we calculated is the deviation of the distance between the correctly identified C_α atoms and the central axial line from 2.5Å distance. RMSD values for the selected proteins are plotted in Figure 6 (on the right).

Table 1. Use of different resolutions

		8Å	**10**Å	**12**Å
1BVP	Correct	100%	89%	78%
	Missed	0	1	2
	C_α	93%	85%	76%
1Q16	Correct	62%	57%	36%
	Missed	20	23	34
	C_α	63%	58%	36%
1TCD	Correct	92%	76%	68%
	Missed	2	6	8
	C_α	85%	77%	66%

Finally, let us underline that the quality of the results is dependent on the resolution of the density maps employed. We tested the program on the density maps at 8Å, 10Å, and 12Å resolutions. Table 1

[a]This is the minimal length of helices detected by default by both *Helix Tracer* and *Helixhunter*.

shows the percentage of correctly identified helices, the number of missed helices, and the percentage of correctly identified C_α. We performed these experiments using the proteins 1BVP, 1Q16, and 1TCD. As expected, the accuracy of *Helix Tracer* degrades as the quality of the density map degrades.

3.2. Comparison with Helixhunter

In this subsection we report the comparison between *Helix Tracer* and *Helixhunter* [7]. We employ the release 1.7 of the *Helixhunter* software. The comparison is performed on the same set of 29 proteins used in the previous subsection. The density threshold used in *Helixhunter* is 0.85, which is the same (*helix-THR*) used in *Helix Tracer*. Although we also tested different density threshold for *Helixhunter*, 0.85 appears to be the threshold that generates best overall results for these 29 proteins.

The comparison starts with the evaluation of the following two parameters (for definition, see the previous subsection):

- *RMSD:* we compare the RMSD values for those C_α atoms that have been correctly identified by both systems;
- C_α*:* we compare the number of C_α that have been correctly identified by either system.

These results are depicted in the two diagrams of Figure 6. *Helix Tracer* correctly identified 75.0% of the total C_α atoms on the helices longer than 8.1Å. *Helixhunter* identified 58.4% of such C_α atoms. For the correctly identified C_α atoms, the RMSD from *Helix Tracer* is on average 0.086Å lower than that of *Helixhunter*. For the protein 1CC5 we reach a peak of improvement of 64% in C_α recognition. Moreover, note that the method adopted in *Helix Tracer* performs always better than *Helixhunter* in terms of the number of correctly identified helices and RMSD.

Figure 7 compares the performance of the two systems in terms of the number of helices that are longer than 5 amino acids. In particular, the diagram on the left compares the number of correctly identified helices (relating them to the number of helices in the PDB annotation), while the figure on the right shows the trend in number of helices present in the PDB annotation and not recognized by either of the systems. Once again, we can observe that *Helix Tracer* provides significantly better results (up to 37% more helices correctly identified) and it never performs worse than *Helixhunter*.

4. CONCLUSIONS AND FUTURE WORK

In this paper, we presented a novel methodology to analyze low resolution density maps (e.g., 6Å to 10Å) of proteins. This is the resolution level that can be obtained for large protein complexes using techniques such as electron cryomicroscopy. At this level of resolution, it is often impossible to recognize the actual backbone directly from the protein density map, but the resolution is sufficient to visually recognize structural features, such as α-helices and β-sheets.

The methodology proposed in this paper makes use of gradients information, extracted from the density map, to perform volumes segmentation and to guide analysis of volumes towards the identification of secondary structure components. In this paper, we focused on the problem of recognizing α-helices. The resulting technique has been implemented in the *Helix Tracer* system, and we performed a test using 29 proteins, with very positive results. In particular, *Helix Tracer* provides a more flexible representation of helices, leading to a more accurate identification.

The outcome of the analysis performed by *Helix Tracer* can be applied to aid in the reconstruction of a tentative atomic model of the entire protein complex. For example, the information about α-helices can be employed as *constraints* to guide and/or filter ab-initio prediction secondary structure, and to aid in threading the protein sequence in the 3D structure. In this direction, we have proposed a framework to map the secondary structures identified from the density map to their locations on the primary sequence of the protein [4]; the framework computes *successful mappings* that best satisfy both the constraints obtained from the density map and the results obtained using secondary structure prediction tools (e.g., PHD). The framework relies on encoding all the components of the problem as a constraints satisfaction problem [9] and makes use of Constraint Logic Programming techniques to solve it.

This gradient-based technique is a general approach, and can be effectively used to recognize other

Table 2. *Helix Tracer* results

ID	Execution Time(s)	PDB Helices	PDB Helices≥ 8.1Å	*Helix Tracer* Identified	*Helix Tracer* Correct	*Helix Tracer* False	*Helix Tracer* Missed	C_α PDB	C_α *Helix Tracer*
1A06	3.5	14	10	10	9	1	1	99	80
1AGW	2.5	8	8	7	7	0	1	98	86
1AXG	3.6	15	10	10	8	2	2	98	61
1B0M	8.9	30	26	33	23	10	3	259	207
1BM1	1.8	7	7	7	7	0	0	161	154
1BRX	1.8	7	7	8	7	1	0	162	156
1BVP	11.7	10	9	10	8	2	1	123	105
1CC5	0.5	4	4	5	4	1	0	41	37
1CI1	5.2	28	25	24	18	6	7	231	172
1DZE	1.9	9	9	8	7	1	2	182	145
1FIY	12.5	40	37	31	28	3	9	572	418
1GIH	2.6	12	11	7	6	1	5	114	76
1JQN	12.3	39	38	35	31	4	7	652	479
1KE8	2.6	13	12	8	8	0	4	121	86
1KGB	1.8	8	8	7	7	0	1	184	168
1M52	11.7	30	20	19	15	4	5	242	183
1NVS	4.9	12	9	10	8	2	1	95	74
1P0C	12.3	34	27	23	17	6	10	240	154
1P14	1.9	14	11	7	7	0	4	122	79
1P8I	1.8	8	7	8	7	1	0	179	166
1Q16	29.0	57	53	37	30	7	23	480	279
1QHD	8.4	14	10	11	8	3	2	122	98
1TCD	5.2	28	25	24	19	5	6	230	177
1TPB	5.0	22	22	17	13	4	9	194	130
2BRD	1.8	7	7	7	7	0	0	169	158
2YPI	5.0	24	24	17	16	1	8	266	158
3PRK	2.0	6	6	11	5	6	1	90	63
7TIM	5.1	24	24	22	18	4	6	216	163
8TIM	5.0	28	22	18	17	1	5	211	150

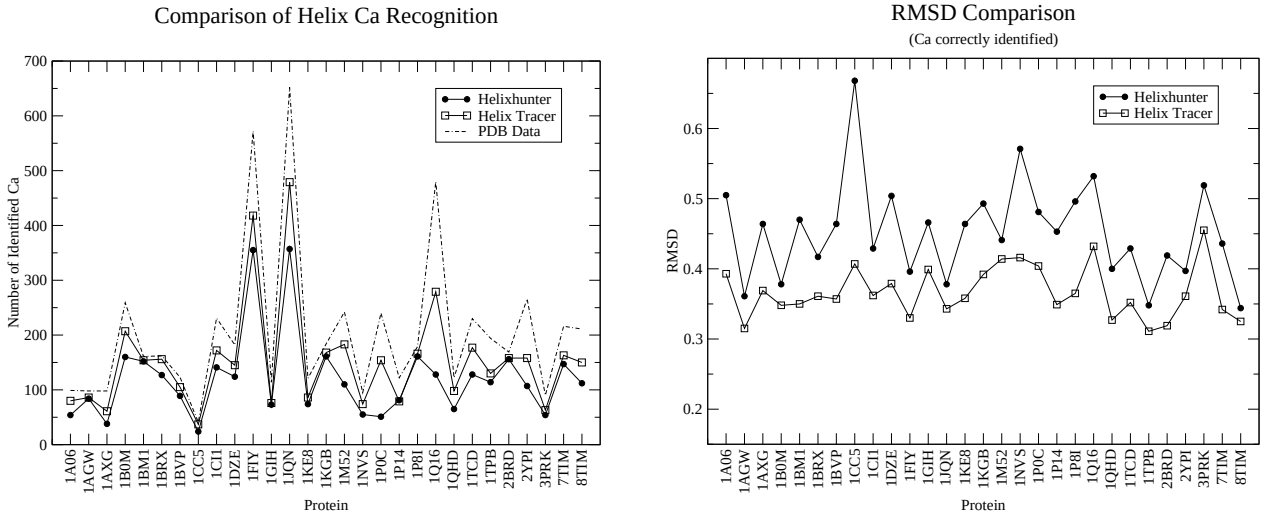


Fig. 6. *Helix Tracer* vs. *Helixhunter*: # of Amino acids identified and RMSD

secondary structure traits of the protein. Work is in progress to apply this methodology to identify β-sheets and coils. Future work will include the development of heuristics to further improve the quality of α-helix identification and to reduce the number of false positives. This will require a more compre-

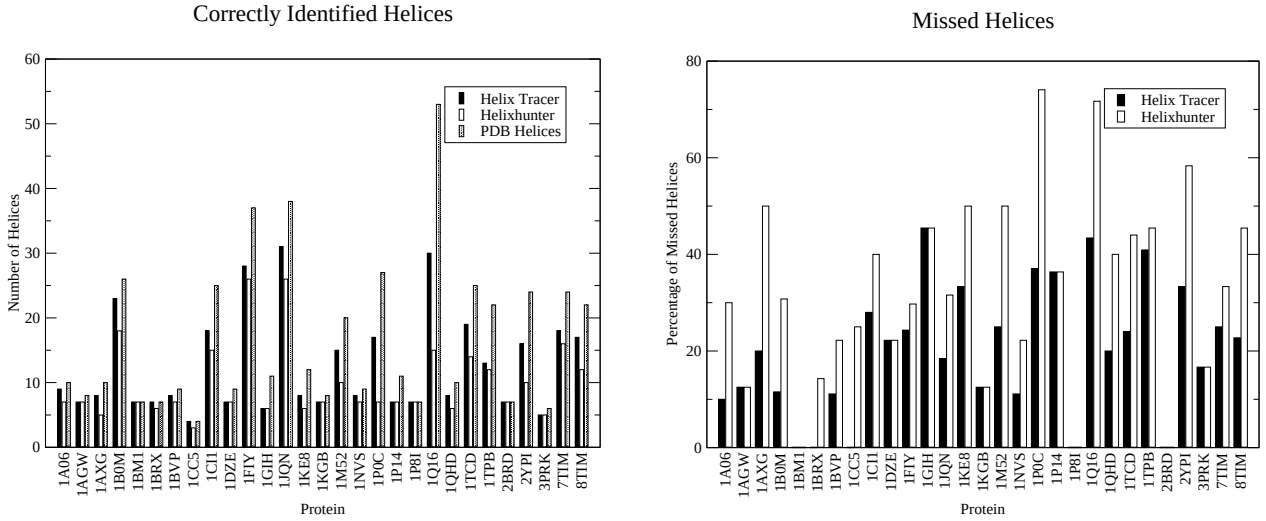


Fig. 7. Correctly identified helices and missed helices

hensive analysis, which will include the recognition of coils and β-sheets. Finally, work is in progress to apply the proposed technique to real data obtained from electron cryomicroscopy.

Acknowledgments

The research has been partially supported by NSF grants CNS-0454066, HRD-0420407, and CNS-0220590.

References

1. R.H. Bartels, J.C. Beatty, and B.A. Barsky. *An Introduction to Splines for use in Computer Graphics and Geometric Modeling.* Morgan Kaufmann Publishers, Los Altos, 1987.
2. B. Böttcher, S.A. Wynne, and R.A. Crowther. Determination of the Fold of the Core Protein of Hepatitis B Virus by Electron Cryomicroscopy. In *Nature*, 386:8891, 1997.
3. W. Chiu et al. Deriving Folds of Macromolecular Complexes through Electron Cryomicroscopy and Bioinformatics Approaches. In *Curr Opin Struct Biol.*, 12:263–269, 2002.
4. A. Dal Palú, E. Pontelli, J. He, and Y. Lu. A Constraint Logic Programming Approach to 3D Structure Determination of Large Protein Complexes. In *ACM Symposium on Applied Computing*, ACM Press, 2006.
5. R. Gonzalez and R. Woods. *Digital Image Processing.* Addison Wesley, 1992.
6. J. Greer. Automated Interpretation of Electron Density Maps of Proteins: Derivation of Atomic Coordinates for the Main Chain. In *J MOl Biol*, 100:427–458, 1974.
7. W. Jiang, M.. Baker, S. Ludtke, and W. Chiu. Bridging the Information Gap: Computational Tools for Intermediate Resolution Structure Interpretation. In *J. of Mol. Biol.*, 308, 2001.
8. Y. Kong and J. Ma. A Structural-informatics Approach for Mining Beta-sheets: Locating Sheets in Intermediate Resolution Density Maps. In *J Mol Biol*, 332(2):399–413, 2003.
9. V. Kumar. Algorithms for CSP: a Survey. In *AI Magazine*, Spring, 32–44, 1992.
10. L. Lo Conte, B. Ailey, T.J. Hubbard, S.E. Brenner, A.G. Murzin, and C. Chothia. SCOP: a Structural Classification of Proteins Database. In *Nucl. Acids Res.*, 28:257–259, 2000.
11. S. Ludtke, P. R. Baldwin, and W. Chiu. EMAN: Semi-automated Software for High Resolution Single Particle Reconstructions. In *J. Struct. Biol.*, 128, 82–97, 1999.
12. E.J. Mancini, M. Clarke, B.E. Gowen, T. Rutten, and S.D. Fuller. Cryo-electron Microscopy Reveals the Functional Organization of an Enveloped Virus. In *Mol. Cell*, 5:255266, 2000.
13. B. Rost. Protein Secondary Structure Prediction Continues to Rise. In *J. Struct. Biol.*, 134:204–218, 2001.
14. C.E. Wang. ConfMatch: Automating Electron-density Map Interpretation by Matching Conformations. In *Acta Crystallogr. D. Biol Crystallogr*, 56:1591–1611, 2000.
15. Z.H. Zhou, M. Dougherty, J. Jakana, J. He, F. Rixon, and W. Chiu. Seeing the Herpesvirus Capsit at 8.5Å. In *Science*, 288:877880, 2000.